LASSO REGRESSION METHOD AND ITS APPLICATION AREAS

Math 490 - Graduation Project

2025-2026 Spring Semester

Author
İrem Simge TURÇ

Supervisor
Assoc. Prof. Dr. Özlem DEFTERLİ

Department of Mathematics

Çankaya University

Abstract

This term project mainly presents the mathematical foundations of Lasso regression, a key supervised learning technique. It also provides illustrative examples from diverse application areas and relevant programming demonstration.

Keywords: Supervised Learning, Lasso Regression, L_1 Regularization

Contents

1 Introduction

1.1 Motivation and Objectives

In today's world of Big Data analytics, analysts often come across large amounts of data in which the number of predictors exceeds that of observations ($p > N$). In such cases, the use of Ordinary Least Squares (OLS) is extremely difficult because OLS is highly sensitive to collinearity and suffers from variance inflation, which results in overfitting and heavy coefficients.

To overcome these limitations, this project focuses on Lasso regression. By utilizing $\mathcal{L}_1$ regularization, Lasso applies a unique mathematical constraint that shrinks coefficients to reduce prediction variance. Unlike Ridge regression ($\mathcal{L}_2$), Lasso can drive the coefficients of irrelevant or redundant variables exactly to zero. This built-in automatic feature selection effectively balances the bias-variance tradeoff, resulting in models that are simpler, highly interpretable, and much better at generalizing to new data.

The primary objective of this graduation project is to provide a comprehensive mathematical and geometric analysis of Lasso regression. Specifically, this study aims to:

- Critically examine the structural failures of the OLS framework when faced with high-dimensionality and strong multicollinearity.
- Compare the parameter estimation and shrinkage behaviors of Lasso ($\mathcal{L}_1$) versus Ridge ($\mathcal{L}_2$) regularization.
- Geometrically illustrate the exact mechanisms that allow Lasso to force coefficients to zero and achieve model sparsity.
- Explain the numerical optimization required for Lasso, with a specific focus on the Coordinate Descent algorithm and the Soft-Thresholding operator.
- Evaluate the practical real-world applications of Lasso and its hybrid extension, Elastic Net, in data-driven fields.

1.2 Structure

The remainder of this graduation project is systematically organized into four subsequent sections to ensure a logical and progressive flow of concepts:

- **Section 2 (Foundations of Linear Models and Regularization):** In this part, we cover the basics of OLS and explain the specific situations where it breaks down. To fix this, we introduce Ridge regression ($\mathcal{L}_2$ regularization) as our first penalized model.
- **Section 3 (Lasso Regression)** This is the main theoretical part of project. I explain how Lasso works mathematically, show its geometric proof for feature selection, and end with Elastic Net to cover Lasso's weak points.
- **Section 4 (Lasso Regression and Application Areas):**This section looks at how Lasso is actually used in the real world. It specifically focuses on its applications in bioinformatics, financial risk modeling, and text mining (NLP).
- **Section 5 (Conclusion):** This final section summarizes the main theoretical takeaways of the project. It evaluates the overall performance of these regularized models and points out potential areas for future research.

1.3 Machine Learning

Machine learning is a subfield of computer science that enables computers to learn from data and get better at tasks without being directly programmed.[1] Machine learning means creating a model with settings that can be changed, then using statistics and optimization to adjust those settings based on training data. For machine learning to be useful, it must make accurate predictions and operate efficiently. Machine learning has transformed numerous fields by facilitating accurate predictions through data analysis. Regression algorithms, a subset of machine learning methods, are extensively employed to predict continuous variables. [2]

Data Quality: The quality of training data directly determines a model's efficiency and accuracy. Since raw data often contains errors from collection or annotation, assessing its quality is a critical step before training. By identifying and fixing these issues early, we can significantly improve model performance and avoid unreliable results.[6]

Data Preprocessing: Real-world data is often messy and full of errors. Preprocessing is an essential step to clean up this data—by handling outliers and missing values—so that our machine learning models can learn accurately and perform well. [6] [7]

Overfitting and Underfitting: Finding the right model complexity requires a delicate balance. If a model is too complex, it memorizes the noise in the training data and overfits; if it is too simple, it misses the main trends and underfits. The ultimate goal is to navigate this bias-variance tradeoff, ensuring the model is just complex enough to learn the underlying patterns and generalize well to new data. [3]

1.4 Supervised Learning

Supervised learning is a machine learning approach driven by pre-labeled datasets. By analyzing sample data paired with the correct outputs (acting as the "ground truth"), the algorithm learns to identify hidden patterns and adjusts its internal parameters to minimize errors. The ultimate goal of this process is to build an accurate model capable of predicting outcomes for entirely new, unseen data. Today, organizations heavily rely on these models to solve complex real-world problems, ranging from filtering spam emails to forecasting stock prices.[3]

Distinction Between Classification and Regression:

Classification and regression are the core algorithms of supervised machine learning and predictive modeling. Both use labeled data to uncover relationships between input features and target outputs, but their goals differ: regression predicts continuous numbers (like house prices), while classification assigns discrete categories (like identifying spam).

Despite the current hype around generative AI, classical supervised learning remains indispensable. Most critical real-world information is still stored in structured, tabular formats. For tasks demanding high precision and clear structure, these algorithms provide immense value across various sectors:

*__Healthcare__: Estimating disease risk (regression) or diagnosing symptoms (classification).

*__Finance__: Calculating loan default probabilities (regression) or detecting fraudulent transactions (classification).

*__Social Science__: Modeling income distributions (regression) or categorizing survey populations (classification).

Beyond predictive accuracy, traditional methods offer two major advantages over complex AI models:

*__Explainability__: Their transparent decision-making is crucial for heavily regulated industries.

*__Efficiency__: They require far less computational power and data, making them perfect for smaller-scale or embedded systems.

Ultimately, regression and classification do not compete with generative AI; they complement it. For example, a classification model can route tasks to the correct generative tool, while a regression model can score the quality of its outputs.[4]

2 Regression of Supervised Learning Method

Regression algorithms are essential in machine learning for predicting continuous variables from independent variables. This project reports a comprehensive analysis of key regression algorithms, examining their strengths, weaknesses, and applications. We present theories, evidence, and supporting data to inform this analysis [1]. Regression methods include Linear Regression, SVM (Support Vector Machine), Random Forest Regression, Ridge Regression, and Lasso Regression.

2.1 Linear Regression

Linear regression is a basic and widely used method in statistical learning. It assumes a straight-line relationship between the independent and dependent variables. In this method, you predict the value of a target variable using one or more predictors. The value you want to predict is called the dependent variable (Y), and the predictors are called independent variables (X). Linear regression finds the best-fitting line or hyperplane by estimating the coefficients of a linear equation, thereby minimizing the difference between the predicted and actual values. People value these models because they are simple, easy to understand, and used across many fields, including biology, psychology, environmental science, social science, and business. [9].

For a scenario with p predictor variables, the theoretical population linear regression model is mathematically defined as:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \varepsilon, \tag{2.1}$$

where β_0 represents the population intercept, β_j (for $j = 1, \ldots, p$) denotes the regression coefficients (slopes) for each independent variable, and ε is the random error term representing the unobserved variation.

2.1.1 Ordinary Least Squares (OLS) Estimation

To estimate the unknown parameters $(\beta_0, \beta_1, \ldots, \beta_p)$ from an empirical dataset, an optimization method is required. Ordinary Least Squares (OLS) is the standard and foundational estimation framework utilized for this purpose. Given a dataset of N observations (x^i, y_i), where $i = 1, 2, \ldots, N$ and $x^i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$, the OLS estimator minimizes the Residual Sum of Squares (RSS). The objective function to be minimized is formulated as follows:

$$RSS(\beta) = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \tag{2.2}$$

To derive the analytical solution for the estimated sample intercept (β_0) within this framework, the partial derivative of the RSS function with respect to β_0 is taken and set to zero:

$$\frac{\partial RSS}{\partial \beta_0} = -2 \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right) = 0 \tag{2.3}$$

By rearranging this equilibrium and dividing both sides by the total number of observations (N), the optimal value of the intercept parameter β_0 is explicitly derived in terms of the sample means of the variables:

$$\beta_0 = \bar{y} - \sum_{j=1}^{p} \beta_j \bar{x}_j \tag{2.4}$$

Here, $\bar{y} = \frac{1}{N} \sum_{i=1}^{N} y_i$ represents the sample mean of the dependent variable, and $\bar{x}_j = \frac{1}{N} \sum_{i=1}^{N} x_{ij}$ represents the sample mean of the j-th independent variable.

2.1.2 Limitations of Ordinary Least Squares and the Motivation for Regularization

Despite its widespread academic application as a Best Linear Unbiased Estimator (BLUE) under standard Gauss-Markov assumptions, the traditional Ordinary Least Squares (OLS) framework exhibits critical methodological vulnerabilities when applied to modern, high-dimensional, or highly correlated data structures. As explicitly identified in the seminal work of Tibshirani [10], the classical OLS estimator suffers from two fundamental limitations that compromise its robustness:

- **Prediction Accuracy and Variance Inflation:** While OLS estimates maintain an unbiased nature, they are highly susceptible to drastic variance inflation under conditions of severe multicollinearity or when the feature space approaches high dimensionality (where the number of predictors p is close to or exceeds the number of observations N). Regularization reduces the magnitudes of the regression coefficients towards zero and even sets some to zero in order to introduce bias and reduce variance.[11].
- **Model Interpretability and Parsimony:** From an analytic perspective, having many predictors means having a high number of coefficients that complicate the understanding of causation or association in relation to the dependent variable. From an empirical perspective, a reduced subset with strong structural influence is very useful. [12].

These systemic constraints serve as the primary mathematical motivation behind the development of penalized regression methodologies, most notably Ridge regression [13] and Lasso (Least Absolute Shrinkage and Selection Operator) regression [10]. By appending a regularization penalty ($\mathcal{L}_2$ or $\mathcal{L}_1$ norms, respectively) directly to the traditional OLS loss function, these techniques explicitly navigate the Bias-Variance Tradeoff. The introduction of this penalty constrains the parameter space, stabilizing the estimation process and systematically improving both the overall prediction accuracy and the generalizability of the predictive model on unseen datasets.

2.2 Ridge Regression ($\mathcal{L}_2$ Regularization)

Ridge Regression is one of the types of regularization techniques that are useful for solving the problems of multicollinearity and overfitting in regular linear regression analysis. It uses an $\mathcal{L}_2$ regularization technique that adds a squared magnitude of coefficients as a penalty to the OLS loss function.

Mathematically, the Ridge regression estimator is defined as the set of coefficients β that minimizes the penalized Residual Sum of Squares:

$$RSS_{Ridge}(\beta) = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{2.5}$$

Where $\lambda \geq 0$ is a tuning parameter that controls the strength of the penalty. When $\lambda = 0$, the penalty term has no effect, and the model reverts to the standard unbiased OLS estimate. As $\lambda \to \infty$, the impact of the shrinkage penalty grows, and the ridge regression coefficient estimates will approach zero [13]. Geometrically, the $\mathcal{L}_2$ constraint forms a smooth, spherical boundary (e.g., a circle in two dimensions). Because this boundary has no sharp corners, the elliptical contours of the unpenalized OLS loss function typically intersect it at non-zero points, meaning Ridge regression shrinks coefficients to reduce variance but does not set them exactly to zero.

3 Lasso Regression

3.1 Mathematical Definition and $\mathcal{L}_1$ Regularization

Lasso (Least Absolute Shrinkage and Selection Operator) regression, introduced by Tibshirani [10], is a powerful alternative to ridge regression. While both address high variance, Lasso employs an $\mathcal{L}_1$ penalty (the sum of the absolute values of the coefficients) rather than the $\mathcal{L}_2$ penalty.

The Lasso objective function minimizes the residual sum of squares subject to this $\mathcal{L}_1$ constraint:

$$RSS_{Lasso}(\beta) = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \tag{3.1}$$

Lasso's major reason for continuous selection of features is because of the difference in $\mathcal{L}_1$ and $\mathcal{L}_2$ norms. In contrast to the spherical shape of Ridge, the constraint region of Lasso is diamond-like polytope with corners along the axis. With the increase in size of OLS loss contours until they touch the constraint region, they usually touch at one of the corners. [11].

Consequently, when the tuning parameter λ is sufficiently large, the $\mathcal{L}_1$ penalty forces certain coefficient estimates (β_j) to become exactly zero. This yields sparse models, meaning Lasso explicitly eliminates irrelevant variables from the final equation. This intrinsic property of automatic feature selection makes Lasso exceptionally advantageous and highly interpretable when analyzing complex, high-dimensional datasets where the number of predictors exceeds the number of observations ($p > N$) [1].

3.2 Working Mechanism of Lasso Regression

Applying Lasso regression follows a series of important conceptual and mathematical steps:[10] [11]

1. **Base Linear Model Formulation:** Lasso starts with a standard linear regression, where the target is predicted as a weighted sum of the input features. Without regularization, this basic model can easily overfit the data.

2. **The $\mathcal{L}_1$-Regularized Objective Function:** Lasso's main idea is to change the usual objective function by adding an $\mathcal{L}_1$ penalty term to the prediction error. The hyperparameter λ controls the strength of this penalty.

3. **Coefficient Shrinkage and Feature Selection:** When λ increases, the estimated coefficients get smaller. Features that matter less are reduced more, and some coefficients become exactly zero.

4. **Embedded Feature Selection:** When a coefficient is zero, it means that the feature is left out. This way, Lasso automatically selects features without needing a separate selection process.

5. **Optimization Strategy:** The non-differentiable nature of the $\mathcal{L}_1$ penalty at zero prevents the use of standard gradient descent. Instead, Lasso is typically optimized using coordinate descent algorithms, which iteratively update one coefficient at a time.

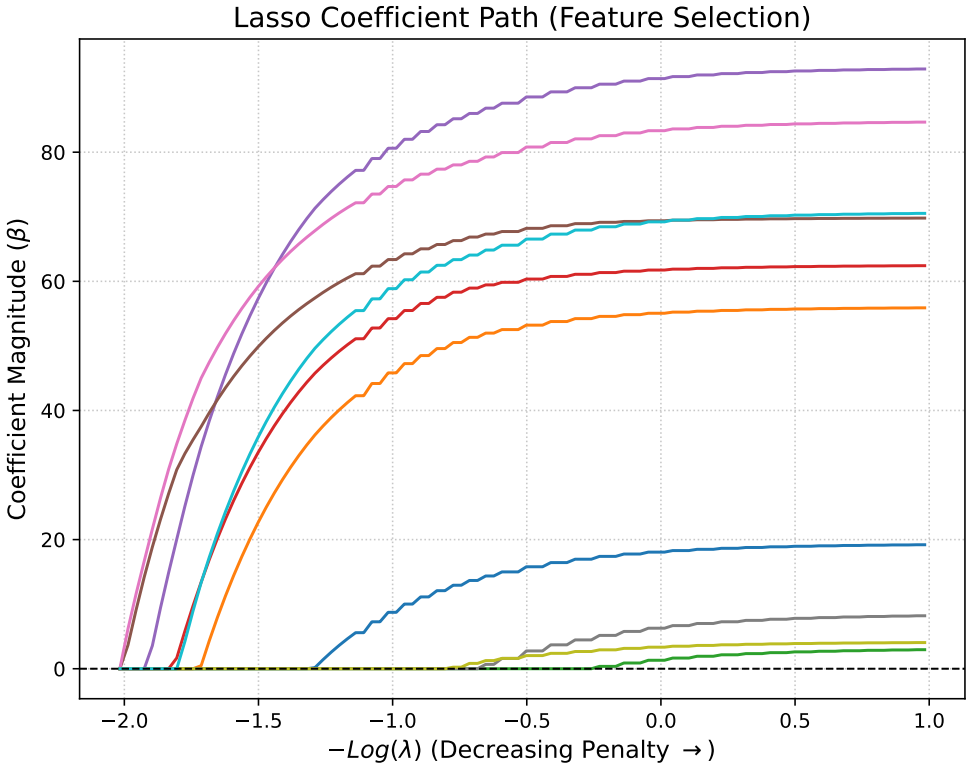

Figure 1: [11]Lasso coefficient path showing how individual feature coefficients shrink exactly to zero as the penalty strength increases, demonstrating automatic feature selection.

3.3 The Bias-Variance Tradeoff in Lasso

The Bias-Variance tradeoff is about finding the right balance between keeping a model simple (bias) and allowing it to capture complex patterns (variance) so it can perform well on new data. Predictive modeling aims to achieve this balance. Lasso Regression helps manage this tradeoff by adjusting its regularization strength, known as lambda (λ):[10]

- **Low λ (Weak Regularization):** When λ is near zero, the penalty is very small. The model fits the training data closely, so it has low bias but high variance. This means there is a higher risk of overfitting to noise.
- **High λ (Strong Regularization):** When λ is large, the model penalizes the coefficients more. Many coefficients are reduced or removed, making the model simpler. This increases bias but greatly reduces variance, often helping the model perform better on test data.

In the end, Lasso does a good job of balancing bias and variance. Removing features can make the model a bit more biased, but the drop in variance usually leads to lower test error and a stronger predictive model.[1].

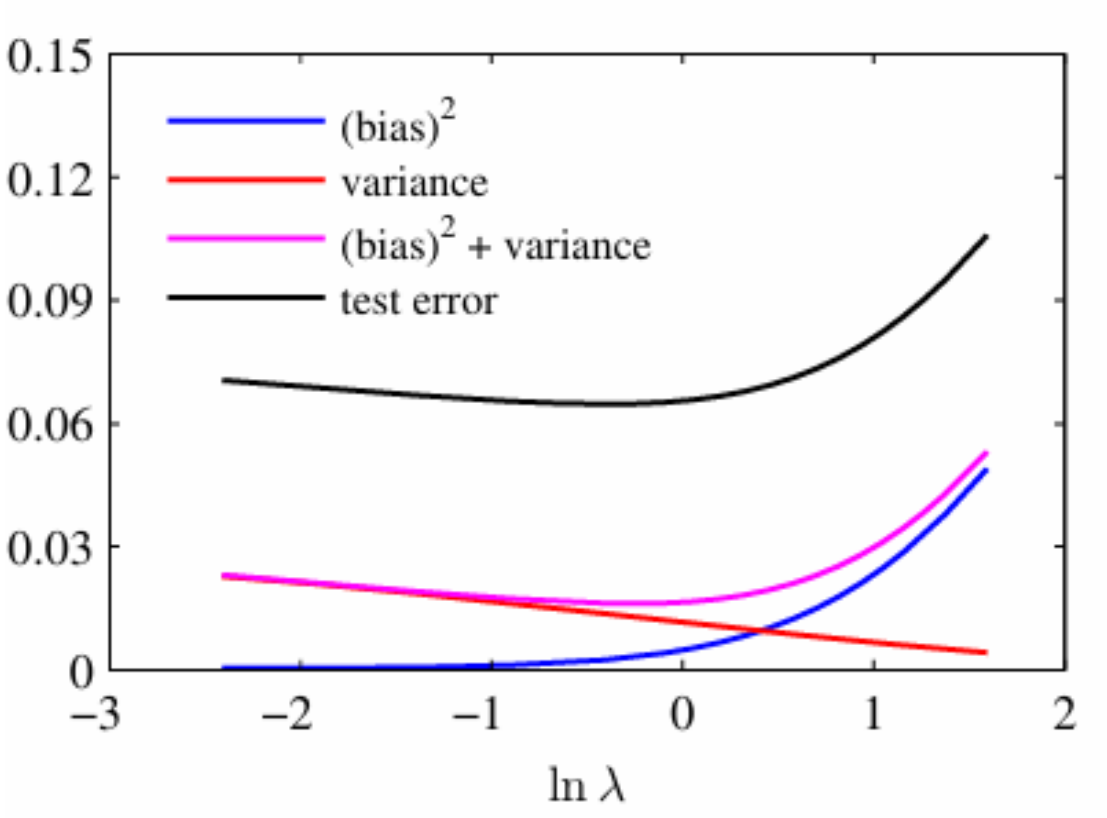

Figure 2: [14]The Bias-Variance tradeoff illustrating the relationship between bias, variance, and test error.

3.4 Selecting the Tuning Parameter λ: Cross-Validation

The performance of Lasso regression depends critically on the choice of the tuning parameter λ. Selecting λ too small causes the model to overfit; selecting it too large causes underfitting by discarding relevant predictors. The standard approach for choosing an optimal λ in practice is **k-fold Cross-Validation (CV)**.

3.4.1 Mean Squared Error (MSE)

Before introducing Cross-Validation, it is necessary to define the error metric it minimises. The **Mean Squared Error (MSE)** measures how far a model's predictions deviate from the true values:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{3.2}$$

where y_i is the true value, $\hat{y}_i$ is the model's prediction for observation i, and N is the total number of observations. [9] The squaring ensures that positive and negative errors do not cancel each other out, and penalises large errors more heavily than small ones.

3.4.2 What is Cross-Validation?

Cross-Validation is a resampling technique used to test the effectiveness of the model on independent datasets. Rather than evaluating the model on the same data it was trained on, CV systematically rotates a held-out validation set through the training data.

Definition 1 (k-Fold Cross-Validation) *: The training dataset of N observations is randomly partitioned into k equally sized, non-overlapping folds $F_1, F_2, \ldots, F_k$. For each fold F_i, the model is trained on the remaining $k-1$ folds and evaluated on F_i. The cross-validated MSE for a given λ is:*

$$\textit{CV-MSE}(\lambda) = \frac{1}{k}\sum_{i=1}^{k} \textit{MSE}_i(\lambda) \tag{3.3}$$

where $\textit{MSE}_i(\lambda) = \frac{1}{|F_i|}\sum_{j\in F_i}(y_j - \hat{y}_j)^2$ is the prediction error on fold F_i when the model is trained on all folds except F_i. [11]

3.4.3 Choosing λ via Cross-Validation

The CV procedure is repeated for each candidate λ over a logarithmic grid of 50–100 values. Two standard selection rules are used:

- $\lambda_{\min}$**:** The value minimising the average CV-MSE. Produces the most predictive model.
- $\lambda_{\mathbf{1SE}}$**:** The largest λ whose CV-MSE is within one standard error of the minimum. Favours a sparser, more parsimonious model.

3.4.4 Why CV Matters for Lasso

Cross-Validation is especially important for Lasso because the sparsity of the resulting model — the number of non-zero coefficients retained — is directly controlled by λ. An incorrectly chosen λ can lead to two failure modes:

- **Underfitting (λ too large):** All or most coefficients are driven to zero. The model loses important predictors and yields high bias.
- **Overfitting (λ too small):** The regularisation is too weak and the model retains noise variables, yielding high variance and poor generalisation.

CV provides a data-driven, objective method for selecting λ that balances these two extremes without requiring external knowledge about the true signal structure.[11].

3.5 Optimization and the Soft-Thresholding Operator

Unlike Ridge regression, which possesses a closed-form analytical solution, the $\mathcal{L}_1$ penalty term in Lasso is non-differentiable when the coefficient is exactly zero. Consequently, traditional calculus-based optimization methods cannot be directly applied. To solve the Lasso optimization problem efficiently, specialized numerical algorithms are required, the most prominent being the Coordinate Descent algorithm [11].

Coordinate descent optimizes the objective function by updating one coefficient β_j at a time while holding all other coefficients constant. By reducing the multidimensional problem into a series of one-dimensional optimization steps, the analytical solution for a single coordinate update is achieved through the Soft-Thresholding operator:

$$S_\lambda(\rho_j) = \text{sign}(\rho_j) \max(0, |\rho_j| - \lambda) \tag{3.4}$$

Where ρ_j represents the simple least squares estimate for the j-th predictor using the partial residuals. The operational logic is straightforward: if the magnitude of the unpenalized coefficient ρ_j is less than the penalty threshold λ, the Soft-Thresholding operator forces the coefficient β_j to become exactly zero. If it is greater than λ, the coefficient is shrunk towards zero by the exact amount of λ. This mechanism is the computational engine that drives Lasso's feature selection capability.

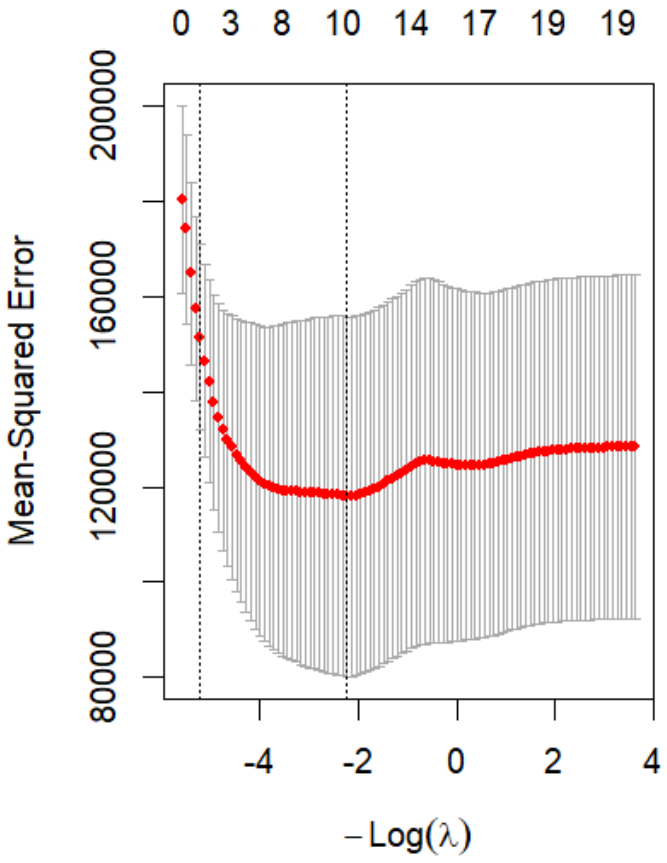

Figure 3: [15]Lasso Cross-Validation curve demonstrating the selection of the optimal penalty parameter (λ) that minimizes the Mean Squared Error.

3.6 Elastic Net Regression ($\mathcal{L}_1 + \mathcal{L}_2$ Regularization)

Although Lasso is highly effective for feature selection, it presents certain limitations. Specifically, when analyzing datasets with a group of highly correlated independent variables, Lasso tends to arbitrarily select only one variable from the group and ignore the rest. Furthermore, in high-dimensional scenarios where the number of predictors exceeds the number of observations ($p > N$), the classical Lasso algorithm can select at most N variables before it saturates.

To overcome these structural constraints, Zou and Hastie (2005) proposed the Elastic Net algorithm [12]. Elastic Net is a hybrid regularization framework that linearly combines the $\mathcal{L}_1$ penalty of Lasso and the $\mathcal{L}_2$ penalty of Ridge regression. The objective function is formulated as:

$$RSS_{ElasticNet}(\beta) = \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2 \tag{3.5}$$

This can also be expressed using a mixing parameter α (where $\alpha \in [0, 1]$) that balances the contribution of each penalty:

$$RSS_{ElasticNet}(\beta) = RSS_{OLS}(\beta) + \lambda \left(\alpha \sum_{j=1}^{p} |\beta_j| + \frac{1-\alpha}{2} \sum_{j=1}^{p} \beta_j^2 \right) \tag{3.6}$$

The inclusion of the $\mathcal{L}_2$ term introduces strict convexity to the optimization problem. This geometric adjustment preserves the automatic feature selection capability of the $\mathcal{L}_1$ norm while enabling the model to retain groups of correlated variables together (known as the "grouping effect"). Consequently, Elastic Net often outperforms Lasso in terms of predictive accuracy when dealing with complex datasets featuring high multicollinearity.

3.7 Geometric Interpretation of Lasso and Sparsity

The most effective way to understand why Lasso regression explicitly forces some coefficients to zero—thereby achieving sparsity and automatic feature selection—is through geometric analysis. The penalized optimization problems of Ridge and Lasso regression can be equivalently formulated as "constrained optimization" problems.

According to this approach, Lasso and Ridge coefficients are the solutions that minimize the classical Residual Sum of Squares (RSS) subject to a specific constraint on the magnitude of the coefficients: [10]

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \tag{3.7}$$

The constraints are defined based on their respective norm types as follows:

$$\text{For Lasso } (\mathcal{L}_1): \quad \sum_{j=1}^{p} |\beta_j| \leq t \tag{3.8}$$

$$\text{For Ridge } (\mathcal{L}_2): \quad \sum_{j=1}^{p} \beta_j^2 \leq t^2 \tag{3.9}$$

Here, t is a constraint threshold that is inversely proportional to the penalty parameter λ used in previous formulas (as t decreases, the penalty increases).

3.7.1 The Geometry of the Constraint Regions

If we examine the geometric shapes of these constraints in a two-dimensional parameter space ($p = 2$ with parameters β_1 and β_2), the structural difference between the two methods becomes visually evident:

- **Ridge Regression ($\mathcal{L}_2$):** The constraint region forms a **circle** (or a hypersphere in higher dimensions) defined by the inequality $\beta_1^2 + \beta_2^2 \leq t^2$. The boundaries of this region are completely smooth and lack sharp corners.
- **Lasso Regression ($\mathcal{L}_1$):** The constraint region forms a **diamond** (or a square) defined by the inequality $|\beta_1| + |\beta_2| \leq t$, with its sharp corners located directly on the coordinate axes.

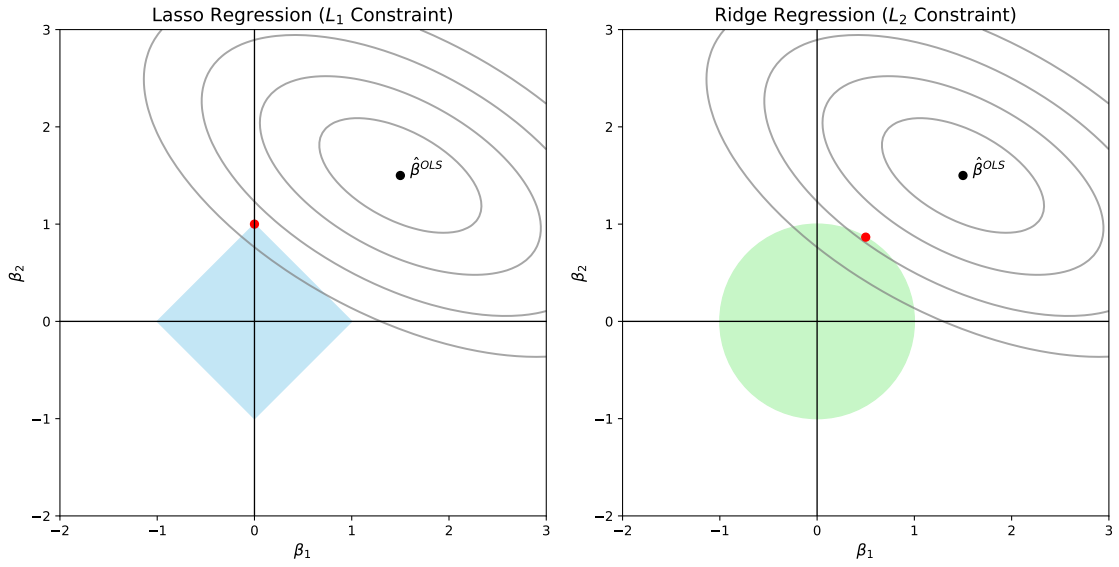

Figure 4: [10]Geometric interpretation of the constraint regions and optimization for Lasso (left) and Ridge (right) regression.

3.7.2 Intersection of RSS Contours and the Constraint Region

Geometrically, the error function (RSS) of traditional OLS creates expanding elliptical contours centered around the unpenalized OLS estimate ($\hat{\beta}_{OLS}$). The main optimization goal is to find the exact point where these expanding ellipses first intersect the penalty constraint region.

In Ridge regression, this constraint region is a smooth circle. Because it lacks sharp edges, the ellipses typically touch the circle somewhere off the axes. This means both coefficients (like β_1 and β_2) shrink but remain non-zero.

In contrast, Lasso's constraint region is a diamond with sharp corners resting directly on the axes. As the RSS ellipse expands, there is a very high probability that it will strike one of these sharp corners instead of a flat edge [11]. When the tangency occurs exactly at a corner, one parameter retains a non-zero value, while the other is forced to exactly zero. This intersecting behavior is the core mathematical and geometric proof of how Lasso naturally achieves sparsity and performs automatic feature selection.

4 Lasso Regression and Application Areas

Lasso regression is valued for its ability to create simple, easy-to-understand models. This makes it a useful tool in many scientific and industrial fields, especially when there are far more features than observations. ($p > N$).

- **Bioinformatics and Genomics:** Lasso is highly effective here because it zeroes out irrelevant biological noise, isolating only the critical genes linked to specific diseases like cancer without overfitting the data.
- **Financial Modeling and Risk Analysis:** In critical operations like credit risk scoring and algorithmic trading, Lasso cuts through this market noise. By selecting only the most reliable predictive factors, it provides financial institutions with the highly transparent and interpretable models required by strict industry regulations.
- **Text Mining and Natural Language Processing (NLP):** Converting text into data (using techniques like TF-IDF) produces massive, mostly empty matrices containing thousands of distinct words. For classification tasks like spam detection or sentiment analysis, Lasso naturally filters out meaningless filler words, keeping only the key vocabulary that actually determines the underlying meaning.

5 Conclusion

With the rise of big data and complex datasets, these become complex, and OLS encounters problems such as high-dimensional spaces, high variances, and multi-collinearity, among others. Therefore, regularization has emerged as a vital method for overcoming these issues.

In this case, we explore the application of lasso regression through use of $\mathcal{L}_1$ penalty. The diamond shape imposed by the $\mathcal{L}_1$ norm has sharp corners along the axis, which cause any contour error of the objective function to shrink at such points, causing some irrelevant coefficients to drop to zero.

Thus, due to this geometric property, Lasso is able to both reduce prediction variance and select significant predictors. In this respect, Lasso regression enables balance in the bias-variance tradeoff and simplicity in model interpretation. By beginning from OLS fundamentals and ending with geometric constraints of the $\mathcal{L}_1$ norm and soft-thresholding operators, this study shows why Lasso and its newer versions are key tools in modern machine learning and predictive analytics.

References

[1] E. Alpaydın, "Introduction to Machine Learning", Second Edition, The MIT Press, 2010.

[2] A. Hasudungan, "A Comprehensive Analysis of Regression Algorithms in Machine Learning," 2024. [Online]. (visited on 04/23/2026). Available: http://andrehasudungan.blog.uma.ac.id/2024/02/05/a-comprehensive-analysis-of-regression-algorithms-in-machine-learning/

[3] IBM. What is supervised learning? (visited on 04/23/2026). Available: https://www.ibm.com/think/topics/supervised-learning#1509394340

[4] IBM. Classification versus regression. Accessed 2025-12-29. Available: https://www.ibm.com/think/topics/classification-vs-regression (visited on 04/23/2026).

[5] Klemczak, B. (2025). Machine Learning-Based Prediction of Heat Transfer and Hydration-Induced Temperature Rise in Mass Concrete. *Energies*, 18(17), 4673.

[6] ACM Digital Library. Available: https://dl.acm.org/doi/10.1145/3447548.3470817

[7] Brijith, Arya. (2023). Data Preprocessing for Machine Learning.

[8] Famili, A., Shen, W. M., Weber, R., & Simoudis, E. (1997). Data preprocessing and intelligent data analysis. *Intelligent Data Analysis*, 1(1), 3-23.

[9] Montgomery, D. C., Peck, E. A., & Vining, G. G. (2021). *Introduction to Linear Regression Analysis* (6th ed.). John Wiley & Sons.

[10] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1), 267–288.

[11] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer Science & Business Media.

[12] Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2), 301–320.

[13] Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1), 55–67.

[14] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006.

[15] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, *An Introduction to Statistical Learning: with Applications in R*, 2nd Edition, Springer Texts in Statistics, Springer, New York, 2021.